REMARKS

In the Office Action of September 2, 2005,[1] claims 3-52 were rejected under 35 U.S.C. § 102(e) as anticipated by U.S. Patent No. 6,085,030 ("*Whitehead*"). Applicants address the rejection below.

Applicants traverse the § 102(e) rejection of claims 3-52 because *Whitehead* fails to support the rejection. In order to properly anticipate Applicants' claimed invention under 35 U.S.C. § 102, each and every element of the claim at issue must be found, either expressly described or under principles of inherency, in a single prior art reference. Further, "[t]he identical invention must be shown in as complete detail as is contained in the . . . claim." *See* M.P.E.P. § 2131. Finally, "[t]he elements must be arranged as required by the claim." *Id.*

With regard to independent claim 3, the Examiner failed to establish that the applied reference teaches each and every feature of the claim. In rejecting claim 3, the Examiner alleged that *Whitehead* discloses "sending the task request to the selected server . . . which downloads any needed byte code . . ., invokes a generic compute technique capable of executing the task request on the selected server and generates results" (Office Action "OA" at 3). This allegation by the Examiner is not supported by *Whitehead*, as discussed below.

Whitehead describes a "network component server architecture" that allows "interaction between a consumer and heterogeneous objects" (Abstract; col. 1, lines 35-52). In alleging that *Whitehead* discloses the above-noted "sending," the Examiner noted *Whitehead*'s disclosure regarding an "object factory repository" and component instantiation (OA at 3; *Whitehead*: col. 10, lines 52-58, col. 11, line 65 - col. 12, line 3). Neither these cited portions nor any other

[1] The Office Action contains a number of statements reflecting characterizations of the related art and the claims. Regardless of whether any such statement is identified herein, Applicants decline to automatically subscribe to any statement or characterization in the Office Action.

portions of *Whitehead* support the Examiner's allegation that *Whitehead* discloses the "sending" feature, as asserted by the Examiner.

Contrary to the Examiner's position, *Whitehead*'s "object factory repository" functionality does not constitute "sending the task request to the selected server . . . which downloads any needed byte code . . ., invokes a generic compute technique capable of executing the task request on the selected server and generates results." *Whitehead*'s object factory repository is "a persistent storage of implementations for the components registered in the offer repository . . . [and] provides the specific object implementation for the offered component" (col. 10, lines 49-53). In *Whitehead*'s system, component requests to the component server can be directed to a "component management service (CMS)" (col. 7, lines 20-24; col. 8, lines 3-5). If the requested component is not available in an "offer repository 254," the CMS forwards the request to an "object factory 240," which "checks the object factory repository . . . to determine if the requested component implementation exists" (col. 8, lines 15-20). Using a repository to obtain component implementations for unregistered components, as disclosed by *Whitehead*, does not constitute "sending the task request to the selected server . . . which downloads any needed byte code . . ., invokes a generic compute technique capable of executing the task request on the selected server and generates results," as asserted by the Examiner.

Furthermore, *Whitehead*'s component instantiation functionality does not constitute "sending the task request to the selected server . . . which downloads any needed byte code . . ., invokes a generic compute technique capable of executing the task request on the selected server and generates results." *Whitehead* describes that the CMS supports "both local and remote instantiation of components" (col. 11, lines 55-61). According to *Whitehead*, local instantiation involves downloading and executing components on the same platform as the requesting

application, while remote instantiation involves loading a local proxy object and binding the remote service to the local proxy object (col. 11, line 61 - col. 12, line 3). Instantiating components locally and remotely does not constitute "sending the task request to the selected server . . . which downloads any needed byte code . . ., invokes a generic compute technique capable of executing the task request on the selected server and generates results."

For at least these reasons, *Whitehead* fails to support the rejection of claim 3 under 35 U.S.C. §102(e). Accordingly, the § 102(e) rejection based on that reference should be withdrawn.

With regard to independent claim 18, the Examiner alleged that *Whitehead* discloses "invoking a generic compute method on the server, which is capable of processing a plurality of types of tasks, which executes the task and generates results." Applicants disagree with the Examiner's characterization of the reference.

In alleging that *Whitehead* discloses the above-noted "invoking," the again Examiner noted *Whitehead*'s disclosure regarding an "object factory repository" and component instantiation (OA at 6; *Whitehead*: col. 10, lines 52-58, col. 11, line 65 - col. 12, line 3). Neither these cited portions nor any other portions of *Whitehead* support the Examiner's allegation. Indeed, using a repository to obtain component implementations for unregistered components, as disclosed by *Whitehead*, does not constitute "invoking a generic compute method on the server, which is capable of processing a plurality of types of tasks, which executes the task and generates results," as asserted by the Examiner. Further, instantiating components locally and remotely does not constitute "invoking a generic compute method on the server, which is capable of processing a plurality of types of tasks, which executes the task and generates results." For at

least these reasons, *Whitehead* fails to support the rejection of claim 18 under 35 U.S.C. §102(e). As such, the § 102(e) rejection of claim 18 should be withdrawn.

As to independent claim 28, the Examiner alleged that *Whitehead* teaches "a computer readable medium containing instructions for controlling a computer system . . . to perform the method of claims 3-17" (OA at 8). Although claim 28 is of different scope than claim 3, Applicants submit that the § 102(e) rejection of claim 28 is unsupported by *Whitehead* and should be withdrawn for at least reasons similar to those presented above in connection with claim 3.

As to independent claim 43, the Examiner alleged that *Whitehead* teaches "a computer readable medium containing instructions for controlling a computer system . . . to perform the method of claims 18-27" (OA at 9). Although claim 43 is of different scope than claim 18, Applicants submit that the § 102(e) rejection of claim 43 is unsupported by *Whitehead* and should be withdrawn for at least reasons similar to those presented above in connection with claim 18.

Claims 4-17 depend upon claim 3; claims 19-27 depend upon claim 18; claims 29-42 depend upon claim 28; and claims 44-52 depend upon claim 43. *Whitehead* fails to support the rejection of claims 4-17, 19-27, 29-42, and 44-52 under 35 U.S.C. § 102(e) for at least reasons similar to those presented above in connection with base claims 3, 18, 28, and 43, respectively. The § 102(e) rejection of claims 4-17, 19-27, 29-42, and 44-52 should therefore be withdrawn. Accordingly, Applicants request withdrawal of the § 102(e) rejection and the timely allowance of pending claims 3-52.

Applicants request the reconsideration and reexamination of this application in view of the foregoing and the timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: December 1, 2005

By: _____
Frank A. Italiano
Reg. No. 53,056